**Visualization and Labelling of Clusters**

* **Cluster Visualization**:
  + PCA was used to reduce dimensionality to 2D for visualization.
  + Scatter plot shows customer segmentation using principal components.
* **Cluster Insights**:
  + **Cluster 0**: Budget-Conscious Short-Term Customers (lower tenure, lower spending).
  + **Cluster 1**: High-Spending Long-Term Customers (higher tenure, higher spending).
* **Visualization Code**:

**from** sklearn.decomposition **import** PCA

**import** matplotlib.pyplot **as** plt

**import** seaborn **as** sns

*# Reduce data to 2D using PCA for visualization*

pca **=** PCA(n\_components**=**2)

X\_pca **=** pca**.**fit\_transform(X)

*# Create a DataFrame with PCA results and cluster labels*

df\_pca **=** pd**.**DataFrame(X\_pca, columns**=**["PCA1", "PCA2"])

df\_pca["cluster\_label"] **=** df["cluster\_label"]

*# Plot the clusters*

plt**.**figure(figsize**=**(10, 6))

sns**.**scatterplot(x**=**"PCA1", y**=**"PCA2", hue**=**df\_pca["cluster\_label"], palette**=**"viridis", data**=**df\_pca, alpha**=**0.7, edgecolor**=**"k")

plt**.**title("Visualization of Customer Clusters (PCA)")

plt**.**xlabel("Principal Component 1")

plt**.**ylabel("Principal Component 2")

plt**.**legend(title**=**"Cluster")

plt**.**grid(**True**)

plt**.**show()

* **Labelling Code:**

*# Analyze cluster characteristics and suggest labels*

cluster\_labels **=** {}

**for** index, row **in** cluster\_summary**.**iterrows():

cluster\_id **=** row["cluster\_label"]

avg\_tenure **=** row["avg\_tenure"]

avg\_monthly\_charges **=** row["avg\_monthly\_charges"]

avg\_total\_service\_cost **=** row["avg\_total\_service\_cost"]

*# Assign labels based on key characteristics*

**if** avg\_tenure **>** 0 **and** avg\_monthly\_charges **>** 0:

label **=** "High-Spending Long-Term Customers"

**elif** avg\_tenure **<** 0 **and** avg\_monthly\_charges **<** 0:

label **=** "Budget-Conscious Short-Term Customers"

**elif** avg\_tenure **>** **-**0.2 **and** avg\_monthly\_charges **>** **-**0.2:

label **=** "Mid-Tier Loyal Customers"

**else**:

label **=** "General Customers"

cluster\_labels[cluster\_id] **=** label

*# Convert cluster label suggestions into a DataFrame*

cluster\_labels\_df **=** pd**.**DataFrame(cluster\_labels**.**items(), columns**=**["cluster\_label", "suggested\_label"])

*# Display clustered data with cluster label and suggested label name*

display(cluster\_labels\_df**.**head())